

This is the author's copy of the publication as archived with the DLR's electronic library at <http://elib.dlr.de>. Please consult the original publication for citation.

# Multiphysical Simulation of a Semi-Autonomous Solar Powered High Altitude Pseudo-Satellite

Reiko Müller and Jane Jean Kiam and Federico Mothes

With advances made in the fields of energy generation from renewable sources, airborne electrical propulsion, and autonomous system operation, much activity has been directed towards the development of so called high altitude pseudo satellites (HAPS) in recent years, with Zephyr (Airbus) and Aquila (Facebook) as prominent examples. Compared to classical orbital satellites, these are designed to require lower deployment costs and to offer a high flexibility in operational tasks and a long mission endurance. In the project StraVARIA, the goal was to develop a high-fidelity multiphysical simulation of such a HAPS, including a long-term mission planner, a reactive guidance system for weather avoidance, a flight control system with protections, a 6-DoF model with solar-electric propulsion system, and a comprehensive environment simulation with 4-D wind and turbulence. Due to the long mission duration, the mission planner and guidance system offer an increased autonomy level compared to standard operator controlled UAVs, however human input is still required for high level planning. The acausal and object-oriented modeling language Modelica has been used to create the integrated simulation model, enabling a modular and detailed modeling approach. By automatic code generation and optimization, simulation efficiency is improved, which is an important factor when considering long-term missions. Results of the integrated simulation show that missions like area surveillance and communication relay are possible whenever adverse weather conditions can be avoided. Ascending to and descending from mission altitude of approximately 18 km also poses a threat to the lightweight HAPS construction since layers of stronger winds and atmospheric disturbance have to be passed. To this end, simulated example missions over Bavaria are presented showcasing these effects, where mission success is ensured by means of the long term mission planner, the reactive guidance, and the inner-level protections implemented in the flight control system.

**Keywords:** aerospace propulsion;artificial satellites;autonomous aerial vehicles;control engineering computing;object-oriented languages;optimisation;program compilers;semiautonomous solar powered high altitude pseudosatellite;energy generation;renewable sources;airborne electrical propulsion;autonomous system operation;high-fidelity multiphysical simulation;long-term mission planner;reactive guidance system;weather avoidance;flight control system;6-DoF model;solar-electric propulsion system;turbulence;long mission duration;automatic code generation;optimization;mission altitude;lightweight HAPS construction;long term mission planner;environment simulation;orbital satellites;object-oriented modeling language;Modelica;unmanned aerial vehicle;size 18.0 km;Object oriented modeling;Atmospheric modeling;Aircraft;Mathematical model;Adaptation models;Meteorology;Control systems

## Copyright Notice

©2018 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE.

## Citation Notice

[1] Reiko Müller, Jane Jean Kiam, and Federico Mothes. Multiphysical Simulation of a Semi-Autonomous Solar Powered High Altitude Pseudo-Satellite. In *2018 IEEE Aerospace Conference*, pages 1–16. Institute of Electrical and Electronics Engineers (IEEE), March 2018. doi:10.1109/AERO.2018.8396531.

```
@INPROCEEDINGS{8396531,
  author={Reiko M{\u}ller and Jane Jean Kiam and Federico Mothes},
  booktitle={2018 IEEE Aerospace Conference},
  title={Multiphysical Simulation of a Semi-Autonomous Solar Powered High Altitude Pseudo-Satellite}},
  organization={Institute of Electrical and Electronics Engineers (IEEE)},
  year={2018},
  volume={},
  number={},
  pages={1-16},
  doi={10.1109/AERO.2018.8396531},
  abstract={With advances made in the fields of energy generation from renewable sources, airborne electrical propulsion, and autonomous system operation, much activity has been directed towards the development of so called high altitude pseudo satellites (HAPS) in recent years, with Zephyr (Airbus) and Aquila (Facebook) as prominent examples. Compared to classical orbital satellites, these are designed to require lower deployment costs and to offer a high flexibility in operational tasks and a long mission endurance. In the project StraVARIA, the goal was to develop a high-fidelity multiphysical simulation of such a HAPS, including a long-term mission planner, a reactive guidance system for weather avoidance, a flight control system with protections, a 6-DoF model with solar-electric propulsion system, and a comprehensive environment simulation with 4-D wind and turbulence. Due to the long mission duration, the mission planner and guidance system offer an increased autonomy level compared to standard operator controlled UAVs, however human input is still required for high level planning. The acausal and object-oriented modeling language Modelica has been used to create the integrated simulation model, enabling a modular and detailed modeling approach. By automatic code generation and optimization, simulation efficiency is improved, which is an important factor when considering long-term missions. Results of the integrated simulation show that missions like area surveillance and communication relay are possible whenever adverse weather conditions can be avoided. Ascending to and descending from mission altitude of approximately 18 km also poses a threat to the lightweight HAPS construction since layers of stronger winds and atmospheric disturbance have to be passed. To this end, simulated example missions over Bavaria are presented showcasing these effects, where mission success is ensured by means of the long term mission planner, the reactive guidance, and the inner-level protections implemented in the flight control system.},
  keywords={aerospace propulsion;artificial satellites;autonomous aerial vehicles;control engineering computing;object-oriented languages;optimisation;program compilers;semiautonomous solar powered high altitude pseudosatellite;energy generation;renewable sources;airborne electrical propulsion;autonomous system operation;high-fidelity multiphysical simulation;long-term mission planner;reactive guidance system;weather avoidance;flight control system;6-DoF model;solar-electric propulsion system;turbulence;long mission duration;automatic code generation;optimization;mission altitude;lightweight HAPS construction;long term mission planner;environment simulation;orbital satellites;object-oriented modeling language;Modelica;unmanned aerial vehicle;size 18.0 km;Object oriented modeling;Atmospheric modeling;Aircraft;Mathematical model;Adaptation models;Meteorology;Control systems},
  ISSN={},
  month={March}
}
```



# Multiphysical Simulation of a Semi-Autonomous Solar Powered High Altitude Pseudo-Satellite

Reiko Müller  
German Aerospace Center (DLR)  
Institute of System Dynamics and Control (SR)  
Münchener Strasse 20  
82234 Wessling, Germany  
reiko.mueller@dlr.de

Jane Jean Kiam  
Universität der Bundeswehr München  
Institut für Flugsysteme  
Werner-Heisenberg-Weg 39  
85579 Neubiberg, Germany  
jane.kiam@unibw.de

Federico Mothes  
Hochschule München  
Fakultät für Flugzeugtechnik  
Lothstr. 64  
80335 München, Germany  
federico.mothes@hm.edu

**Abstract**—With advances made in the fields of energy generation from renewable sources, airborne electrical propulsion, and autonomous system operation, much activity has been directed towards the development of so called high altitude pseudo satellites (HAPS) in recent years, with *Zephyr* (Airbus) and *Aquila* (Facebook) as prominent examples. Compared to classical orbital satellites, these are designed to require lower deployment costs and to offer a high flexibility in operational tasks and a long mission endurance. In the project *StraVARIA*, the goal was to develop a high-fidelity multiphysical simulation of such a HAPS, including a long-term mission planner, a reactive guidance system for weather avoidance, a flight control system with protections, a 6-DoF model with solar-electric propulsion system, and a comprehensive environment simulation with 4-D wind and turbulence. Due to the long mission duration, the mission planner and guidance system offer an increased autonomy level compared to standard operator controlled UAVs, however human input is still required for high level planning. The acausal and object-oriented modeling language *Modelica* has been used to create the integrated simulation model, enabling a modular and detailed modeling approach. By automatic code generation and optimization, simulation efficiency is improved, which is an important factor when considering long-term missions. Results of the integrated simulation show that missions like area surveillance and communication relay are possible whenever adverse weather conditions can be avoided. Ascending to and descending from mission altitude of approximately 18 km also poses a threat to the lightweight HAPS construction since layers of stronger winds and atmospheric disturbance have to be passed. To this end, simulated example missions over Bavaria are presented showcasing these effects, where mission success is ensured by means of the long term mission planner, the reactive guidance, and the inner-level protections implemented in the flight control system.

## TABLE OF CONTENTS

REFERENCES .....	1
1. INTRODUCTION.....	1
2. MULTIPHYSICAL SIMULATION MODEL .....	2
3. FLIGHT CONTROL SYSTEM.....	6
4. MISSION PLANNER.....	7
5. WEATHER AVOIDANCE MODULE .....	9
6. INTERFACING AND PERFORMANCE .....	10
7. SIMULATION RESULTS .....	11

8. SUMMARY AND OUTLOOK.....	14
ACKNOWLEDGMENTS .....	14
REFERENCES .....	14
BIOGRAPHY .....	16

## 1. INTRODUCTION

Increasing autonomy levels has been a prevalent topic in the design of Unmanned Aerial Vehicles (UAVs) since the beginning of the 90's. The well-known 3-D (dull, dirty and dangerous) notion describes activities or missions which drive the development of many autonomous systems, not only in the aerospace field. The main goal is to reduce human interaction in tasks

- involving repetitive actions while not requiring a lot of attention and input (for example reconnaissance missions or communication relay),
- performed in unsafe environments where human presence requires complex technical solutions (for example contaminated areas),
- considered as high-risk missions / dangerous (i.e. operations in war theatres).

We focus on the special case of a solar-powered UAV operating in stratospheric altitudes. Such a platform combines satellite-like abilities with a less complex and cost-intensive system (in Figure 1, two examples are depicted, with the already operational Airbus *Zephyr* platform in Figure 1a, and the DLR research prototype *ELHASPA* in Figure 1b). This system specification is particularly challenging, as it requires flying at physical limits with an aircraft that needs to be highly optimized in several aspects:

- A long endurance is key to mission success, which is why conventional, non-regenerative propulsion systems, requiring refueling and extensive maintenance, are ruled out. Instead, a regenerative and purely electric approach is more appropriate: The conventional propellers are driven by electric motors and the photo-voltaic panels are attached to the wings as energy conversion units, while the excess is stored in lithium-ion battery packs. The overall propulsion system has to be very efficient, since the specific power output is far lower than that from burning fuel.
- As a direct consequence, the aircraft needs to have a very lightweight structure, so that it can reach stratospheric



(a) Airbus Zephyr High Altitude Pseudo Satellite (HAPS)



(b) DLR ELHASPA prototype aircraft

**Figure 1:** Examples for solar-powered high altitude aircraft platforms

altitudes with limited power and stay there for a long period of time. A compromise has to be found between weight and airframe stability, as well as operation speed and payload. Such an aircraft therefore has a very restricted flight and loads envelope.

- The previously mentioned properties also make a High Altitude Pseudo Satellite (HAPS) highly dependent on external effects like temperature and solar irradiation, as well as susceptible to weather phenomena like wind and turbulence. Unlike conventional aircraft, flights and trajectories can only be flown in areas with favorable weather conditions (calm winds), and at latitudes where shorter nights can be sustained without recharging the battery. This poses high demands on a guidance system since it should provide safe operation and weather avoidance, optimal mission execution and operation at a high autonomy level for a slow and fragile aircraft.

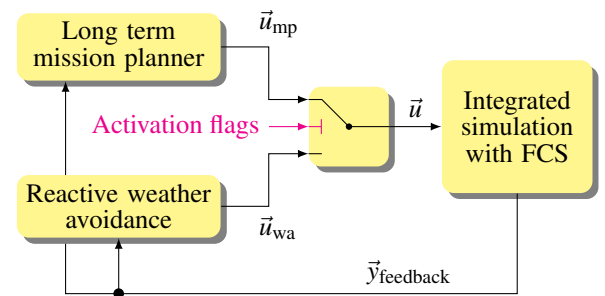
Despite these difficulties, several attempts and successful flights of HAPS and High Altitude Long Endurance (HALE) systems were performed in the course of the last ten years with the most prominent examples from Airbus<sup>2</sup>, Facebook<sup>3</sup> and recently China's Aerospace Science and Technology Corporation (CASC)<sup>4</sup>. However, the fact that currently only three HAPS have performed flight tests, underlines the nontrivial nature of designing a such a system. An overview and discussion of various platforms can be found in [1] and [2] for instance.

### The StraVARIA project

System simulation is essential for the development and operation of a HAPS platform, since structural and operational safety margins are especially tight. It is therefore impor-

tant to provide accurate models for each design stage and during operation, to increase overall system safety again. The StraVARIA project targets to provide a conceptual study of a HAPS system, focusing on an increase in autonomy regarding mission planning and aircraft guidance. Steps towards this goal include the definition of requirements for HAPS and weather avoidance systems, the investigation of regulatory aspects, as well as the ability to certify such an aircraft, which was worked out by Airbus Defence and Space (ADS). The development of weather detection sensors, generation of a weather awareness system, weather avoidance strategies, energy optimal operation management through flight control and mission planning, and development of communication data links were carried out respectively by partners Airbus Group Innovations (AGI), Universität der Bundeswehr München (UniBW), Hochschule München (HM) and the DLR Institutes of Atmospheric Physics (DLR-PA) and System Dynamics and Control (DLR-SR). One of the project outputs is an integrated simulation model in the state of a Technology Readiness Level (TRL) two to three, which means usage for simulation studies in lab environments. Implementation-ready systems for flight hardware and architectures are therefore not developed, however advanced graphical user interfaces for the long term mission planner were realized for example, in order to enable realistic operator training and mission planning. Likewise, the reactive weather avoidance is designed to ultimately yield real time capability on an onboard guidance computer. Lastly, the control system and aircraft model is designed to be easily exportable to hardware platforms, by using the Functional Mockup Interface (FMI) standard [3].

In this paper, we focus on the description of the integrated simulation model, consisting of the multiphysical simulation of the aircraft with its environment and weather (Section 2), flight control system (Section 3), long-term mission planner (Section 4) and the reactive weather avoidance system (Section 5). These constituting parts are depicted in Figure 2, with the actual combination to a common simulation setup discussed in Section 6. It is subjected to two simulated HAPS mission scenarios detailed in Section 7. A summary of the results and an outlook for coming work are finally given in Section 8.



**Figure 2:** StraVARIA simulation setup with the long term mission planner and the reactive weather avoidance working as guidance modules for the integrated simulation model. The aircraft input  $\vec{u}$  is selected according to an activation logic which selects the planner outputs depending on the aircraft's condition and also planner flags (e.g. weather dependent).

## 2. MULTIPHYSICAL SIMULATION MODEL

Accurate simulation models are an integral part of an aircraft design process where model complexity changes with the

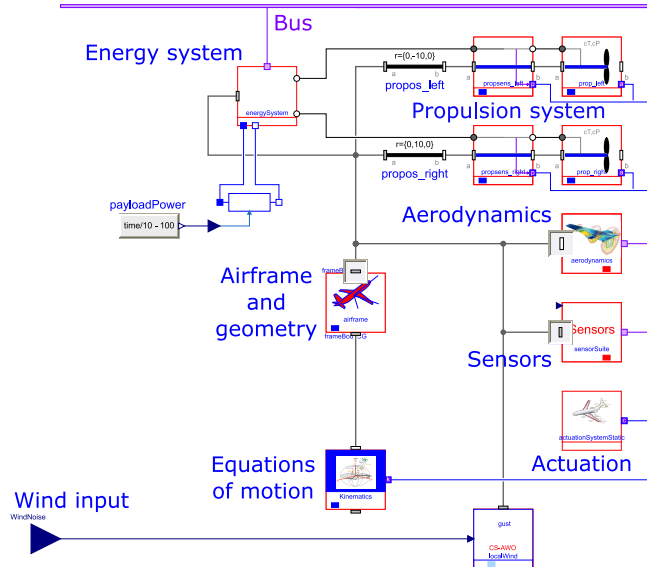
<sup>2</sup><http://defence.airbus.com/portfolio/uav/zephyrhaps/>

<sup>3</sup><https://code.facebook.com/posts/268598690180189>

<sup>4</sup><http://www.janes.com/article/71772/solar-electric-cai-hong-uav-conducts-stratospheric-flight>

respective development stage. During the conceptual phase, simplified and empirical modeling methods allow to generate approximate performance figures like range and masses, which are important to judge the capabilities for the desired mission types. In subsequent steps, the modeling complexity is increased, for example generating a rigid-body model for baseline controller design, an FEM model for structural analysis, or a CFD model for calculation of aerodynamic forces. Iterating through the design stages can be regarded as an optimization problem with the design variables as degrees of freedom, requirements as constraints, and specific performance measure(s) as criteria. Multidisciplinary Design Optimization (MDO), specifically for aerodynamic and structural analysis, is a research topic at the DLR as well, motivating several projects (for example *Digital-X*, [4]), to bring competences from various institutes together into an optimized design process.

A modular or object-oriented modeling approach for the aircraft, subsystems and environment models is therefore advantageous. Variants of and inheritance from one integrated model can be used, to be adapted easily for different applications by changing respective modules or their level of detail and also increasing consistency in the overall design process. In *StraVARIA*, the modeling language *Modelica* [5] was applied to model the aircraft, the systems and the physical environment, with some aircraft components either adapted or simplified from a previous DLR HAPS system including flexible structural dynamics implemented by Klöckner et al. [6]. The control system, and the interfacing to the guidance tools are developed in *Modelica* as well. This language implements the so called "acausal" modeling paradigm, which allows the user to state the model or the dynamics as a set of equations (in opposition to classical algorithmic modeling languages like *MATLAB*, where the "flow" direction of the variable needs to be defined). The equations are reordered automatically (by the IDE *Dymola*) to transform them into a set of differential (algebraic) equations, which can then be integrated during time simulation. By applying this modeling approach, the block-based diagram layer of the aircraft model shown in Figure 3 is obtained.



**Figure 3:** Block-based aircraft model in *Dymola* IDE adapting the modeling approach from the DLR FlightDynamics [7] and MultiBody libraries [8].

### Equations of motion

The *StraVARIA* aircraft is modeled as a rigid body, since detailed mass and geometric data was not available, hence no elastic deformation and structural loads analysis could be performed. Some basic geometric definitions are given in Table 1. The rigid body model has twelve differential equa-

**Table 1:** Basic dimensional data for the *StraVARIA* airplane.

Parameter	Variable	Value
Aircraft length	$l_{ac}$	7.7 m
Wingspan	$b$	32 m
Wing area	$S$	64 m <sup>2</sup>
Mean chord	$c_{mac}$	2.2 m

tions for the three translational and three rotational degrees of freedom. To describe the propagation of the aircraft's center of gravity position over the Earth, the kinematic equations are applied:

$$\begin{aligned}\dot{\phi} &= \frac{v_N}{R_n(\phi) + h}, \\ \dot{\lambda} &= \frac{v_E}{(R_e(\phi) + h) \cos(\phi)}, \\ \dot{h} &= -v_D,\end{aligned}\quad (1)$$

with  $R_n$  and  $R_e$  being meridional and prime-vertical radii, latitude  $\phi$ , longitude  $\lambda$ , altitude  $h$  and the inertial velocity in North East Down (NED) coordinates  $\vec{V}_K = [v_N, v_E, v_D]$ . When taking into account wind, the true airspeed is introduced:

$$\vec{V}_K = [v_N, v_E, v_D] = \vec{V}_{tas} + \vec{V}_{wind}. \quad (2)$$

The aircraft moves due to the sum of forces acting on the center of gravity according to Newton's second law:

$$\sum \vec{F} = \frac{d(m\vec{V}_K)}{dt}. \quad (3)$$

Since there is no mass change due to the electric propulsion system,  $\frac{dm}{dt} = 0$ , and the equation can be further expanded to:

$$\sum \vec{F} = m(\vec{a}_{b,e} + \vec{a}_{transport} + \vec{a}_{coriolis}), \quad (4)$$

with indices  $b, l, e$  and  $i$  referring to the aircraft body-fixed, aircraft-carried local NED, Earth-centered-Earth-fixed and the Earth-centered-inertial coordinate systems and  $\vec{r}, \vec{v}$  and  $\vec{a}$  denoting the position, velocity and accelerations respectively. The transport and coriolis accelerations are given by:

$$\vec{a}_{transport} = \vec{a}_{e,i} + \dot{\vec{\omega}}_{e,i} \times \vec{r}_{l,e} + \vec{\omega}_{e,i} \times (\vec{\omega}_{e,i} \times \vec{r}_{l,e}), \quad (5)$$

$$\vec{a}_{coriolis} = 2\vec{\omega}_{e,i} \times \vec{v}_{l,e}, \quad (6)$$

and are small in this application (low airspeeds). The sum of external moments  $\vec{M}$  acting on the aircraft's center of gravity are determined by:

$$\sum \vec{M} = \frac{d(\mathbf{I}_b \vec{\omega}_{b,i})}{dt} = \frac{d(\mathbf{I}_b(\vec{\omega}_{b,l} + \vec{\omega}_{l,e} + \vec{\omega}_{e,i}))}{dt}, \quad (7)$$

with the aircraft inertia tensor  $\mathbf{I}_b$  and its components  $I_{xx}, I_{xy}, I_{xz}, I_{yy}, I_{yz}$  and  $I_{zz}$ . The two latter angular velocities  $\vec{\omega}_{l,e}$  and  $\vec{\omega}_{e,i}$  are small and are neglected for the coming derivation (but not in the actual simulation model).



The sum of forces  $\vec{F}$  is comprised of aerodynamic forces (lift and drag), engine thrust, as well as the weight force. The aerodynamic forces and moments are defined in the aerodynamic coordinate system, which is aligned with the velocity vector of the aircraft relative to the surrounding air. The engine thrust  $\vec{T} = [X_T, Y_T, Z_T]$  and moments are defined in the aircraft body fixed system, taking into account off center angles for the thrust vector, while the weight is defined in the local NED system. These forces have to be transformed in body axes and then used in Equation (4). The force equations with respect to the  $l$  - system are given by:

$$\begin{aligned} \dot{u}_b &= r_b v_b - q_b w_b - g \sin \Theta + (X_A + X_T) / m, \\ \dot{v}_b &= -r_b u_b + p_b w_b + g \sin \Phi \cos \Theta + (Y_A + Y_T) / m, \\ \dot{w}_b &= q_b u_b - p_b v_b + g \cos \Phi \cos \Theta + (Z_A + Z_T) / m, \end{aligned} \quad (8)$$

with angular rates  $\vec{\omega}_{b,l} = [p_b, q_b, r_b]^T$  and the velocity components  $\vec{v}_{b,l} = [u_b, v_b, w_b]^T$  resolved in the body fixed frame, obtained from the  $l$  frame by rotation around the Euler angles  $\Phi$ ,  $\Theta$  and  $\Psi$ . When assuming an aircraft symmetric to its body  $x-z$  plane ( $I_{xy}, I_{yz} = 0$ ), the moment equations can be simplified to:

$$\begin{aligned} L &= I_{xx} \dot{p}_b - I_{xz} (\dot{r}_b + p_b q_b) + (I_{zz} - I_{yy}) q_b r_b, \\ M &= I_{yy} \dot{q}_b + I_{xz} (p_b^2 - r_b^2) + (I_{xx} - I_{zz}) p_b r_b, \\ N &= I_{zz} \dot{r}_b - I_{xz} (\dot{p}_b - q_b r_b) + (I_{yy} - I_{xx}) p_b q_b. \end{aligned} \quad (9)$$

The aerodynamic forces ( $X_A, Y_A, Z_A$ ) and moments ( $L, M, N$ ) needed for the solution of (8) and (9), depend on the dynamic pressure  $\bar{q}$  and the aerodynamic coefficients ( $C_x, C_y, C_z, C_L, C_M, C_N$ ):

$$\begin{aligned} X_A &= \bar{q} S C_x, & Y_A &= \bar{q} S C_y, & Z_A &= \bar{q} S C_z, \\ L &= \bar{q} S b C_L, & M &= \bar{q} S \bar{c} C_M, & N &= \bar{q} S b C_N. \end{aligned} \quad (10)$$

The controls are in this case the three control surfaces for ailerons  $\delta_a$ , elevator  $\delta_e$  and rudder  $\delta_r$ , as well as the throttle setting  $\delta_T$ . Ailerons are deflected in opposing fashion, while elevators and engine throttle are equally deflected.

### Energy system

The energy and propulsion system shown in Figure 4 consists of the following components:

- dual electric motors with motor controllers and flanges to attach the propellers,
- photo-voltaic panels attached to the top of the wing to collect the solar radiation,
- and a simplified battery model, which integrates all power demands and contributions of engines, solar panels and payload.

The engines are controlled by a PI motor controller which converts throttle setting or RPM to a torque demand and a respective input current. A non-adjustable propeller with constant thrust coefficient  $c_T$  is attached to the motor with an efficiency rating of  $\eta_{prop} = 0.9$ . The thrust  $T_{prop}$  and torque  $Q_{prop}$  of the propeller are then given by:

$$T_{prop} = c_T \cdot \rho_a \omega^2 D_{prop}^4, \quad (11)$$

$$Q_{prop} = c_Q \cdot \rho_a \omega^2 D_{prop}^5 = T_{prop} \cdot D_{prop} \frac{c_P}{c_T}, \quad (12)$$

with ambient air density  $\rho_a$ , the propeller diameter  $D_{prop}$ , shaft angular velocity  $\omega$  and torque coefficient  $c_Q$ .

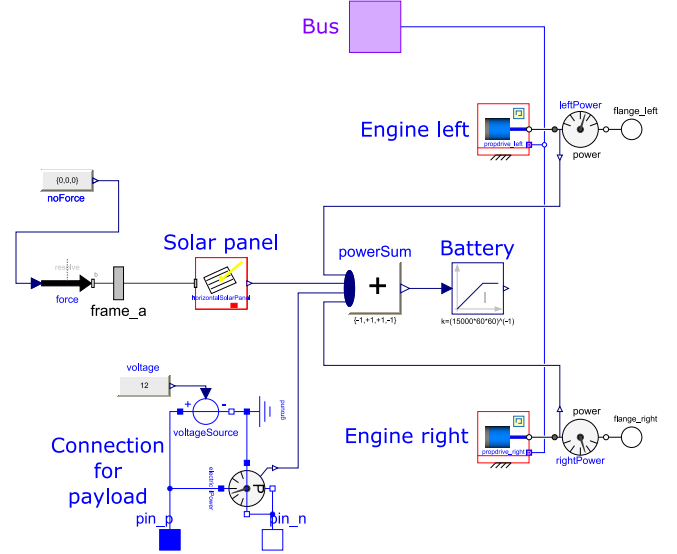


Figure 4: Stravaria energy system based on blocks from the Modelica Standard Library.

The propeller power demand can be calculated via:

$$P_{prop} = 2\pi\omega Q_{prop} = 2\pi\rho_a c_Q \omega^3 D_{prop}^5. \quad (13)$$

The battery is a limited integrator (with the upper limit set at 100 % state of charge), which is initialized with an energy content given at simulation start. It then integrates the sum of the power demands of the motors and payloads and the contribution of the solar panels. This energy is then subtracted from or added to the energy charged in the batteries.

The solar panels have a fixed conversion efficiency of  $\eta_{cell} = 14\%$ , and the solar radiation is calculated in dependence of the Julian date and the position over the Earth affecting local time, sunrise, and sunset. As a simplification, it is assumed that the maximum power can always be retrieved from the panels. This is normally achieved by using a maximum power point tracker adapting the panel voltage accordingly, which however has been omitted for this application to improve simulation efficiency. Furthermore, losses due to atmospheric effects are considered in power output as given in equation (14), with the maximum solar irradiance being  $E_{max} = 1366 \frac{W}{m^2}$ .

$$P_{output} = E_{max} \cdot f_{geo} \cdot f_{atmo} \cdot f_{cloud} \cdot A_{panel} \cdot \eta_{cell}. \quad (14)$$

The geometry factor  $f_{geo}$  is cosine-shaped with its maximum at midday. The atmosphere factor  $f_{atmo}$  is dependent on the mass of the air  $m_{air}$  above the observer (equaling one at  $h = 0$  and 0 at  $h \rightarrow \infty$ ). When taking into account the ground maximum filtered irradiance  $E_{max,flt} = 1120 \frac{W}{m^2}$ , it can be calculated via:

$$f_{atmo} = \left( \frac{E_{max}}{E_{max,flt}} \right)^{m_{air}}. \quad (15)$$

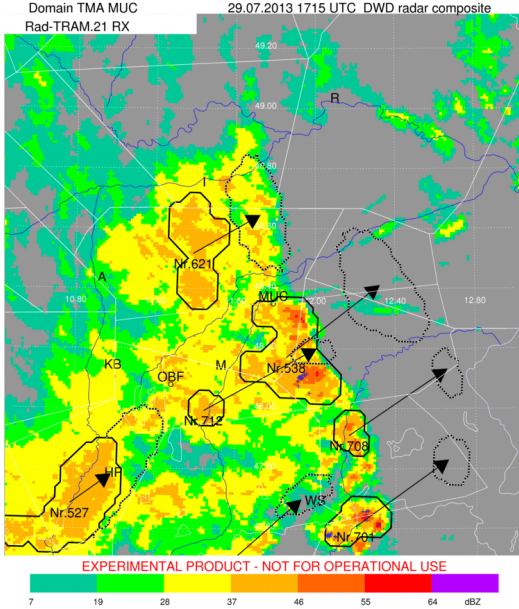
The cloud factor finally is calculated by:

$$f_{cloud} = 1 - 0.75 \cdot c_{cover}^3, \quad (16)$$

with the cloud coverage  $c_{cover}$  ranging between 0 (no clouds) and 1 (completely overcast).

**Table 2:** List of available weather data for the integrated simulation and their usage in the modules long-term mission planner (MP), reactive weather avoidance (WA) and integrated simulation model (Sim).

Weather	Data source	Algorithm	Type	Forecast time / steps	Update steps	Format	Used in
Precipitation	Radar	Rad-TRAM	Polygon	0-1 hour / 5 minutes	5 minutes	XML	WA
Thunderstorm	Satellite	Cb-TRAM	Polygon	0-1 hour / 15 minutes	5 minutes	XML	WA
Thunderstorm	COSMO-DE	Cb-LIKE	Polygon	1-6 hours / 1 hour	1 hour	XML	MP
Atmospheric variables	COSMO-DE	-	Scalar	0-21 hours / 1 hour	3 hours	GRIB	MP, WA, Sim



**Figure 5:** 60 min nowcast by the algorithm Rad-TRAM over Bavaria. Black lines are detected precipitation cells  $\geq 37$  dBZ, dotted lines show the nowcast for  $t_0 + 60$  min.

#### Weather data for mission planning and simulation

In the *StraVARIA* project, historical weather data for the different modules was provided by DLR's Institute of Atmospheric Physics, with an overview given in Table 2. This includes the forecast as well as the corresponding measurement over Europe for three days in 2015 with significant weather activity. For the nowcast (0 – 1 h) of heavy precipitation cells and thunderstorms, the data of the two algorithms Rad-TRAM (Radar TRacking and Monitoring) [9] and Cb-TRAM (Cumulonimbus TRacking And Monitoring) [10] are used, mainly in the reactive weather avoidance module. As indicator for clear air turbulence (CAT), the Richardson-Number ( $Ri$ ) is provided. Since the long-term mission planner needs forecasts farther ahead in time, thunderstorm cell prediction for a timespan of 1 – 6 h is realized by means of the Cb-LIKE (Cumulonimbus LIKElihood) algorithm [11]. The previously mentioned data is available as comprehensive weather situation map in eXtensible Markup Language (XML) [12], which can be easily processed in the respective planners. The three dimensional windfield forecast (0 – 21 h) from the COSMO-DE model is provided in GRIdded Binary (GRIB) format, and is considered as the "real" weather inside the planners and the integrated simulation (details are given in the next section).

On the basis of the algorithm Rad-TRAM, the concept of the so called No-Go areas (NGAs) can be explained: The algorithm detects and tracks heavy precipitation cells and also issues a nowcast. Although in nature transitions are fluid,

thresholds such as radar reflectivity can be used to bound regions of danger. Every detected cell is marked with a unique identification number, represented as polygon and designated as NGA (see Figure 5). The application of polygons is intuitive, little memory consuming and allows fast processing. Concepts like G-AIRMETS (Graphical AIRman's METeoro-logical Advisories) prove that a geometric representation of weather leads to an improved weather situational awareness in pilots, with better go/no-go decisions [13]. In the presented concept, NGAs are semipermeable geometries (it is only possible to exit, but not to penetrate NGAs), whose form and position are time-variant. The spatial expansion of an NGA can be adapted for every aircraft, as it solely depends on the respective toughness. Hence these properties differentiate NGAs from static and definite No-Fly zones.

**Listing 1:** 4-D weather data interpolation model in *Modelica*.

```

model gribExtractor_4d_stravaria
  "Extracts variable data from a grib file (for StraVARIA with levelType=
  150)"
algorithm
  // Access the grib-file only when the aircraft moves outside of the
  // current four-dimensional box defined by the 16 vertices of the
  // box in variables [lat, lon, h, t]

  when { initial() ,
    not (la_cosmo >= min(pre(outlats)) and la_cosmo <= max(pre(
    outlats))),
    not (lo_cosmo >= min(pre(outlons)) and lo_cosmo <= max(pre(
    outlons))),
    not (altitude >= min(pre(altitudeLevels)) and altitude <= max(pre(
    altitudeLevels))),
    not (time24h >= min(pre(timePts)) and time24h <= max(pre(
    timePts))) } then
    // Find the enclosing grid points from the GRIB data definition in time
    // and altitude

    // With these, four calls to function grib_nearest of GRIB_API per
    // variable provides the 16 points of the 4d polytope

  end when;

equation

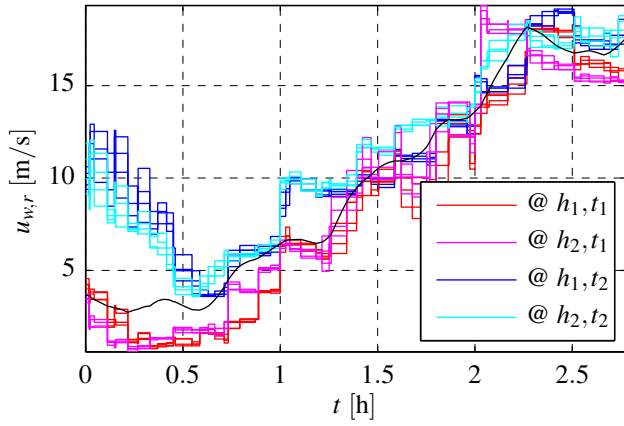
  // With the 16 points the Vandermonde matrix is built and solved by
  // LAPACK routines yielding the interpolated value at the current 3
  // D position and time

end gribExtractor_4d_stravaria;

```

#### Environmental modeling for the integrated simulation

The environment has been modeled using the capabilities of DLR's FlightDynamics [7] and Environment libraries [14]. These implement the definitions of WGS-84 ellipsoid along with the EGM-96 gravitational model. The solar irradiation is calculated according to the equations mentioned in Section 2. The climatic forecast model COSMO-DE, developed by the German weather forecasting service DWD [15], is employed for realistic modeling of wind and turbulence in the simulation. The variables of the precalculated forecast are stored in a GRIB file, and are then interpolated during



**Figure 6:** Interpolated zonal wind velocity  $u_{w,r}$  from COSMO-DE for a climb from an altitude of 0 to 10 km. 4-D grid points are shown by means of the  $u$ -values at the four positions, two grid altitude levels  $h_1, h_2$  and two grid time steps  $t_1, t_2$  surrounding the current 4-D position.

simulation using a dedicated *Modelica* model (see Listing 1). As COSMO is defined on a regular grid in latitude, longitude, altitude and time<sup>5</sup>, a linear 4-D interpolation scheme can be employed to obtain values for longitudinal, lateral and vertical winds as well as the Turbulent Kinetic Energy (TKE). In order to improve simulation efficiency, the GRIB file is only queried, when the aircraft passes a grid boundary (either in latitude/longitude, altitude or time). By using the `pre()` operator in *Modelica*, it is possible to detect and generate an integration event at these boundaries, obtain new nearest values from the GRIB file, and interpolate either in the new values or the old ones otherwise. The interpolation is stated as a linear problem using the Vandermonde matrix and can be solved with standard linear algebra packages for the current variable value(s). The resulting interpolation is shown in Figure 6 for the zonal wind component  $u_{w,r}$ , where each step in the boundary variables indicates an event iteration of the integrator that is triggered if one argument of the `when()` clause becomes true (see Listing 1). From the GRIB file, the zonal and meridional winds  $u_{w,r}$  and  $v_{w,r}$ , the vertical wind  $w_w$  and the turbulent kinetic energy  $k$  are retrieved. As COSMO is defined in a rotated coordinate system, the wind components need to be transformed into local geodetic coordinates  $u_{w,g}, v_{w,g}$  according to:

$$\begin{aligned} u_{w,g} &= u_{w,r} \cos \delta + v_{w,r} \sin \delta, \\ v_{w,g} &= -u_{w,r} \sin \delta + v_{w,r} \cos \delta, \end{aligned} \quad (17)$$

with  $\delta$  as angle between the meridians between the geographic  $(\varphi, \lambda)$  and the rotated system  $(\varphi_r, \lambda_r)$ :

$$\delta = \arctan \left( \frac{\cos \varphi_r \sin(\lambda_r - \lambda)}{\cos \varphi \sin \varphi_r - \sin \varphi \cos \varphi_r \cos(\lambda_r - \lambda)} \right). \quad (18)$$

Inclusion of continuous turbulence is realized by noise spectra shaped according to the Dryden turbulence model [16]. The turbulence intensities for the three wind components  $\sigma_u$ ,  $\sigma_v$  and  $\sigma_w$  are however calculated from the TKE:

$$k = \frac{1}{2}(\sigma_u^2 + \sigma_v^2 + \sigma_w^2), \quad (19)$$

<sup>5</sup>The cell width is  $\approx 2.8$  km in the horizontal plane, while in the vertical direction 50 levels from 10 – 21500 m with altitude differences of 20 – 1000 m are given. These 3-D messages are stored for every hour, thus forming the resulting 4-D weather data array.

which is assumed to be isotropic, hence:

$$\sigma_{u,v,w} = \sqrt{\frac{2}{3}} \cdot k. \quad (20)$$

The filter transfer functions can then be formulated with the scale lengths  $L_u$  and  $L_v$  for high altitudes as given in the specification:

$$\begin{aligned} H_u(s) &= \sigma_u \sqrt{\frac{2L_u}{\pi V_{\text{tas}}}} \frac{1}{1 + \frac{L_u}{V_{\text{tas}}} \cdot s}, \\ H_{v,w}(s) &= \sigma_v \sqrt{\frac{2L_v}{\pi V_{\text{tas}}}} \frac{1 + \frac{\sqrt{3}L_v}{V_{\text{tas}}} \cdot s}{\left(1 + \frac{L_v}{V_{\text{tas}}} \cdot s\right)^2}. \end{aligned} \quad (21)$$

These are connected to white noise generating blocks provided by the Modelica Noise library [17]. When assuming median TKE  $\approx 0.02$  and TAS  $\approx 30 \frac{\text{m}}{\text{s}}$  values, the cutoff frequency for the transfer functions (21) can be determined. Even with a sampling frequency five times higher than the cutoff frequency, a relatively large sampling time of  $\approx 10$  s is obtained. This in turn is beneficial for the overall simulation time, as less integration events are generated by the white noise process. The resulting filter outputs are then added to the wind transformed in body coordinates (Equation (17)) and applied to the equations of motion (Equation (2)).

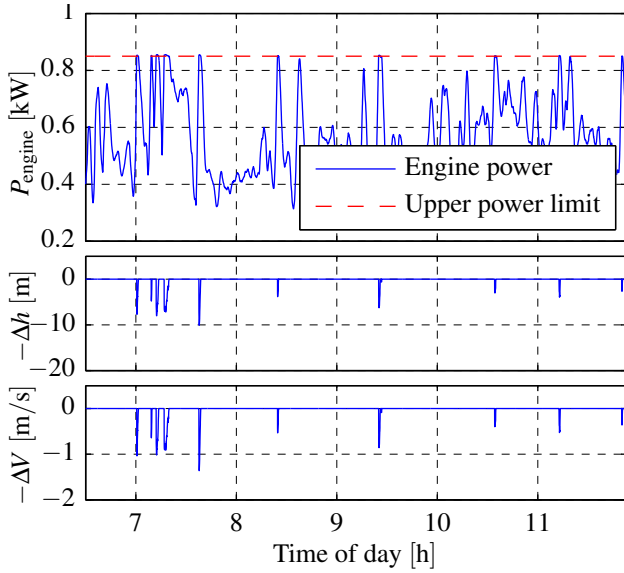
### 3. FLIGHT CONTROL SYSTEM

The main task of the flight control system is to keep the aircraft on the desired trajectory, as well as preventing unstable flight states and damping external disturbances. As shown when describing the equations of motion, going from aircraft controls as moment generating devices to the actual trajectory reference (position) involves several integrations. Furthermore, parts of these equations depend on aircraft properties directly, while the trajectory states for instance are purely kinematic and independent of the aircraft. Another simplification for fixed-wing aircraft is that the longitudinal and lateral dynamics can be decoupled. This allows to separately design controllers for longitudinal and lateral channels. A fixed-structure approach for the FCS has therefore been taken, yielding the classical cascaded controller layout with inner loops for mode stabilization and damping and outer loops or autopilot for trajectory tracking. This is achieved by translating eventual deviations from commanded altitude, speed, track, and sideslip angle into corrections for the aircraft's orientation angles and throttle. For the longitudinal channel, the Total Energy Control System (TECS) concept was selected, which is based on the rationale that changes in altitude and velocity are coupled, as they cause changes in potential and kinetic energy. Based on the x-component of the equations of motion in vehicle carried flight path coordinate system:

$$m\dot{V}_k = T - D - mg \cdot \sin \gamma_k, \quad (22)$$

trimmed horizontal flight is assumed, and errors in  $\dot{V}_k$  and  $\gamma_k$  are canceled with thrust and pitch commands  $\Delta T$  and  $\Delta \Theta$ , respectively. TECS is hence designed to use the thrust as energy generating control, and the elevator for energy distribution, details are given in the original publication [18] and adaptations [19]. Likewise, the lateral autopilot channel is designed according to the Total Heading Control System (THCS) layout, whose working principle is similar to that of TECS. It transforms deviations from commanded track and





**Figure 7:** Engine power limited to  $P_{\max} = 850$  W by Pseudo Control Hedging (PCH) controller through adding correction commands  $\Delta h$  and  $\Delta V$  to the reference altitude  $h_c$  and velocity  $V_c$  during mission scenario described in Section 7.

sideslip angles into roll and yaw rate commands. For the inner loop controller, standard implementations of pitch, roll and yaw dampers are used, with gains specifically tuned to aircraft type and mission scenarios.

#### Flight envelope limitation

The *StraVARIA* HAPS can only operate inside tight operational bounds, which is why the probability of exceeding these is rather high, even when considering only calm winds. To cope with this problem, limiters are implemented throughout the controller (as for roll angle), and for throttle, a pseudo-control hedging approach is implemented (see [20] for the application to the optimization of trajectories for passenger aircraft). Limiting thrust and the speed of the aircraft with respect to the surrounding air (taking into account wind) works by adapting the reference commands for altitude and true airspeed or inertial speed (for 3D / 4D trajectory tracking respectively). For the minimum equivalent airspeed, the assumption of 6 m/s is drawn, from which the minimal  $V_{\text{tas,min}} = V_{\text{eas,min}} \cdot \sqrt{\rho_{\text{isa}}/\rho_a}$  is calculated by using ambient and main sea level air densities ( $\rho_a$  and  $\rho_{\text{isa}}$ ). The upper limit for  $V_{\text{eas}}$  is set to 25 m/s, which is an estimated value from previous experiences with similarly light-weight solar airplanes [6]. For the considered scenarios however, the equivalent airspeed is held at a constant value of 9 m/s to fly with the optimum angle of attack, yielding minimum drag. With the airspeed being the critical measure, it is given priority over path following when saturations occur. The *prio* - variable in the PCH controller shown in Figure 8 can be set to a value between zero and one, equating to altitude and velocity priority respectively. If the actual velocity exceeds its limits, *prio* will be set to velocity priority ( $= 1$ ), meaning that reference altitude changes are also used to adjust velocity. Furthermore excess of the thrust limitations are fed back with variable  $\Delta T$ , which is transformed into an altitude and velocity correction command ( $\Delta T > 0 : \rightarrow h_{\text{lim}} < h_c, V_{\text{lim}} < V_c, \Delta T < 0 : \rightarrow h_{\text{lim}} > h_c, V_{\text{lim}} > V_c$ ) by the PCH. This is based on the relation given in (22) as well, after performing a linearization around the current flight state and

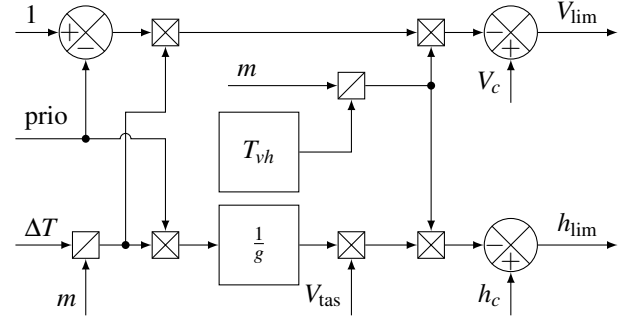
assuming constant drag ( $\Delta D = 0$ , which is valid for small time differentials) and small angles  $\sin \gamma \approx \gamma$ , one obtains:

$$m\Delta V = \Delta T - mg \cdot \Delta \gamma. \quad (23)$$

With  $\Delta h \approx V\Delta \gamma$ , the defining equation for the controller can be stated:

$$(1 - \text{prio}) \cdot \frac{\Delta T}{m} + \text{prio} \cdot \frac{\Delta T}{m} = \Delta V + g \frac{\Delta h}{V}, \quad (24)$$

and the actions of the thrust limiter during the scenario of Section 7 are shown in Figure 7. This setup serves the main purpose of a safety limitation, which can react quickly to external effects like wind gusts that cannot be avoided when using weather forecasts for planning. It therefore complements the reactive weather avoidance module, and is the innermost safety system.



**Figure 8:** Controller to hedge the altitude and velocity references ( $h_c$  and  $V_c$ ) according to thrust limits. The time constant  $T_{vh}$  is necessary to reduce the amount of stiffness in the closed-loop system as well as avoidance of chattering.

## 4. MISSION PLANNER

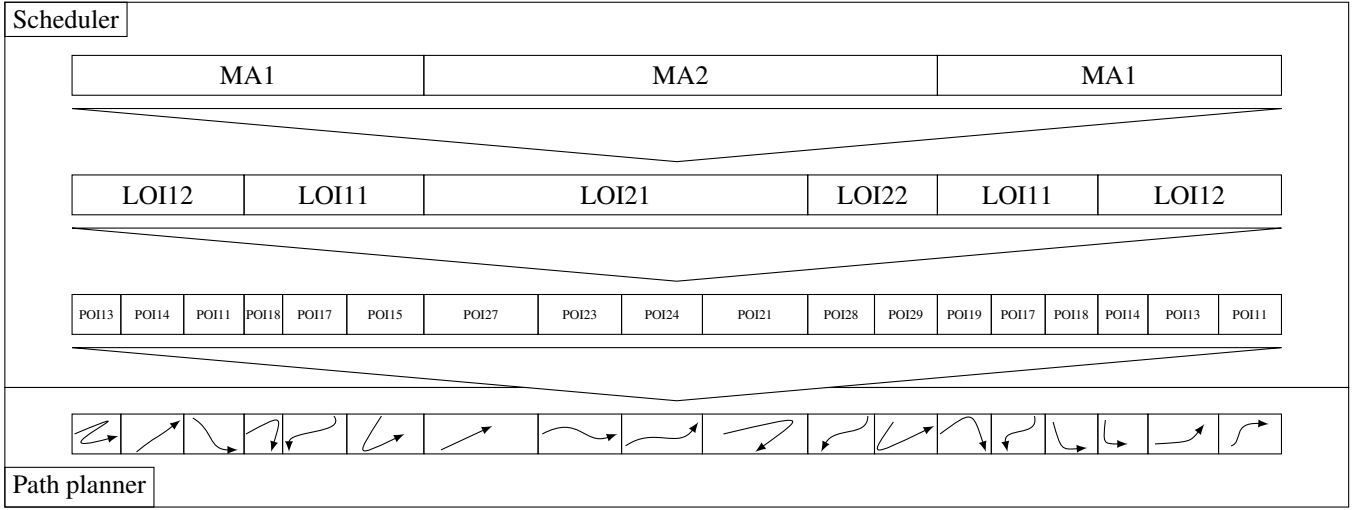
In this section, we focus on the system design of the multilateral planner for long-term missions to be carried out by a single HAPS. Unlike conventional aircraft, a HAPS is, as mentioned earlier, technically more challenging due to:

- the rather insignificant propulsive power of the electric motors that make it difficult to follow given paths, especially in a wind field,
- the long-term mission spanning over hours or days in a substantial 3-dimensional airspace, which increases the complexity of the search for a feasible optimal solution.

A hybrid approach involving scheduling and planning is very widely used, for example in [21] and [22]. The scheduler decides logically for the execution order of the tasks, while the lower level planner optimizes the way the tasks are executed. Kiam and Schulte adapted the hybrid architecture to a probabilistic problem [23]. The scheduler first proposes several schedules with different task execution orders. Subsequently, the analytical planner calculates the UAV paths for the schedules that most probably maximize the mission rewards within the allocated planning time. Once the planning time is due, the schedules, that are not yet considered by the analytical planner will be ignored.

#### Logical Planning

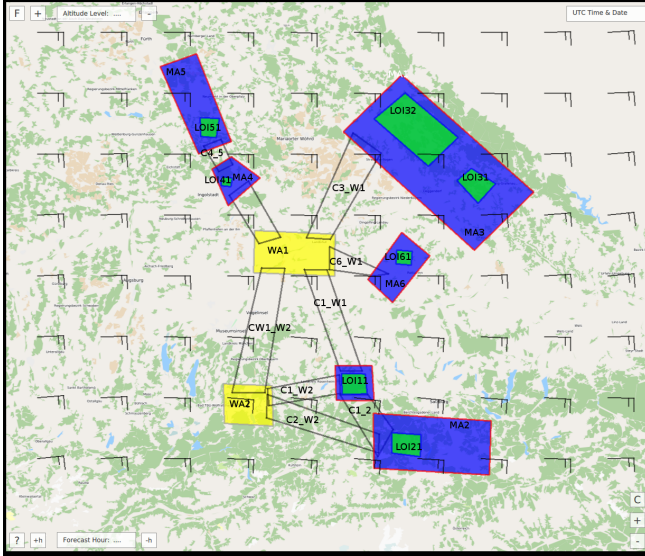
Figure 10 shows a simple exemplary mission scenario. The Locations Of Interest (LOIs) are areas to monitor, while the



**Figure 9:** Hierarchical scheduling and planning for an exemplary set of two mission areas with associated LOIs and POIs.

Mission Areas (MAs) are allocated airspace, within which the HAPS is allowed to move during mission execution. The HAPS commutes between MAs through corridors (C). In this typical mission, the HAPS operates at a flight altitude of 18 km and is equipped with a low energy consumption electro-optic camera to carry out its surveillance tasks. The mission success rate depends therefore on the cloud coverage. In order to determine schedules with the best probable mis-

sion order of POIs of each LOI is determined (see Figure 9). The scheduler proposes multiple feasible schedules. The feasibility of a schedule is governed by the airspace allocated by the air traffic controller, the coverage of critical weather zones due to weather and also mission requirements, for example time to monitor (using the available weather forecast as mentioned in Section 2 and as detailed in [11] and [25]). Since the mission success rate is probabilistically correlated to the cloud coverage, the problem is modeled as a Markov decision process and the ranking of the proposed schedules is determined by the computed expected rewards.



**Figure 10:** An exemplary surveillance mission scenario in which the HAPS is contracted to monitor the locations of interest as often as possible

sion rewards, the scheduler has to consider the weather (cloud coverage, wind vectors, and zones with critical weather) in each mission element. In [23], the scheduler is conceived in a hierarchical manner: the different abstraction levels are ordered according to the dimensions of the mission elements, namely MA, LOI and Point Of Interest (POI). Similar to a hierarchical task network (HTN) [24], the higher level task is decomposed into lower level tasks, i.e. the surveillance mission in our example is first decomposed into MAs, and each MA is decomposed into an order of LOIs, and subsequently

#### Analytical Flight Path Planner

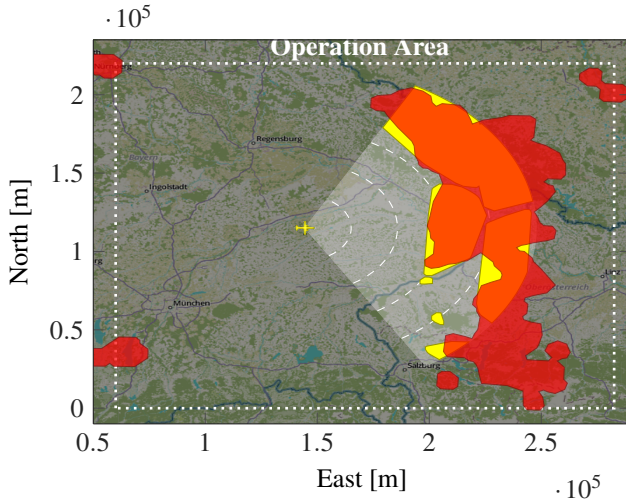
Schedules generated by the logical scheduler are passed to the analytical flight path planner, in which the consecutive POIs are considered as start and goal points. Since not all paths are feasible for a HAPS because wind cannot be easily compensated by stronger propulsion, it is hence important to employ an analytical planner, that is capable of taking into consideration the kinodynamic constraints, as well as the time-varying environment constraints (airspace allocation and critical zones). By employing the path planning method described in [26] for discrete planning problems, the effects of the control inputs of the vehicle, as well as the wind effects on the kinematics of the HAPS can be easily taken into account. The kinematics of the HAPS considered by the flight path planner are described using Equation (1) with a spherical Earth assumption (meridional and prime-vertical radii  $R_n$  and  $R_e$  are set to the median Earth radius  $R \approx 6378$  km). An action-based discrete path planner considers a set of actions, while searching for an optimal path. In our case, we use a set of feasible discrete turn rates  $A_\psi = \{-|\psi_{\max}|, -|\psi_{\max}| + \epsilon_\psi, \dots, |\psi_{\max}| - \epsilon_\psi, |\psi_{\max}|\}$  and pitch angles  $A_\theta = \{-|\theta_{\max}|, -|\theta_{\max}| + \epsilon_\theta, \dots, |\theta_{\max}| - \epsilon_\theta, |\theta_{\max}|\}$ , where  $\epsilon_*$  denotes the discretization step, to account for the kinematic constraints of the HAPS. The authorized operation airspace, as well as the time-varying critical weather zones are considered in a discrete path planner as physical constraints or obstacles. The flight path planning problem can be solved either using a control-based planner like the RRT planner from the Open Motion Planning Library [27], or a domain independent planning like the ENHSP [28].

### Selection of the Best Plan

As mentioned above, the scheduler computes several feasible schedules and ranks them based on the expected mission reward. The ranking is however probabilistic and inaccurate due to the innate probabilistic property of the weather forecasts and the approximated time needed for each higher level subtask in the hierarchical scheduler. The latter contributes to the inaccuracy of the computed reward and cost of each scheduler. More accurate rewards and costs can be computed with the analytical path planner. Therefore, the final ranking can only be determined at the lowest abstraction level of the hierarchical scheduler/planner. Although considering the forecasted weather conditions and the flight kinematics complicating the planning, the selected plan is of better quality in the sense that the feasibility and reward are optimized, which in turn results in less frequent replanning. Replanning occurs only, if the weather conditions differ too much from the forecast, or if the mission requirements are altered.

## 5. WEATHER AVOIDANCE MODULE

The University of Applied Sciences Munich (HM) works on a module for HAPS that automatically generates flight paths to overcome adverse weather conditions. The objective is to increase their automation and safety levels, and by this means improve mission fulfillment. In civil aviation, approximately 20% of the accidents are related to adverse weather [29]. Fragile aircraft like HAPS are especially at risk, and therefore require a particularly provident avoidance in certain phases of the flight, i.e. ascent and descent through weather active troposphere [30]. The focus lies on the handling of thunderstorm, turbulence and wind for a short-term span up to one hour.

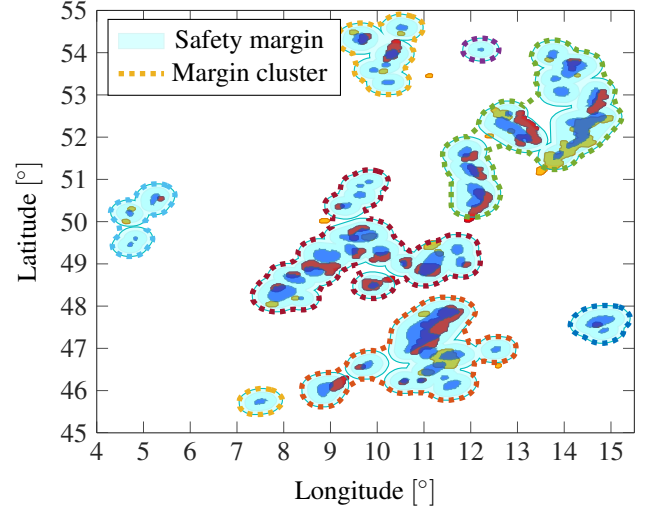


**Figure 11:** Implementation of a simulated onboard radar by Langmeier and Stenzel (AGI): Thunderstorm nowcast ● and onboard radar ● can differ considerably. Additional NGAs are considered for the avoidance.

### Challenges for weather avoidance

Commonly, pilots use onboard weather radar to detect and avoid thunderstorms. A radar displays the current situation regarding precipitation, but does not offer a prediction of future weather development. For a pilot in a fast flying airliner, this actual snapshot still holds enough information to

find a way through adverse weather. This is possible because the aircraft groundspeed is significantly higher than that of a thunderstorm. For the pilot, the weather is quasi-static, and he uses safety margins to prevent unwanted contact. As HAPS are very light with limited energy capacity, radar systems are in most cases too heavy and energy consuming for them. HAPS are generally also very slow, which is emphasized by the fact that they sometimes will even fly backwards with respect to Earth because flight performance cannot compensate for wind. For this special kind of aircraft, weather cannot be treated as static. The avoidance module hence depends on information by external sources for an anticipating flight planning [30]. Finding an optimal path in a time-variant environment in the presence of uncertainties automatically is a tough challenge. The short lifespan of convective weather like thunderstorms poses a particular challenge. It must be clear, that in case of unpredicted weather changes, it is not possible to even find a nearly optimal path. The optimality of a path directly depends on the closeness between available forecast and actual weather. In retrospect, a better path could have been found in almost every case.



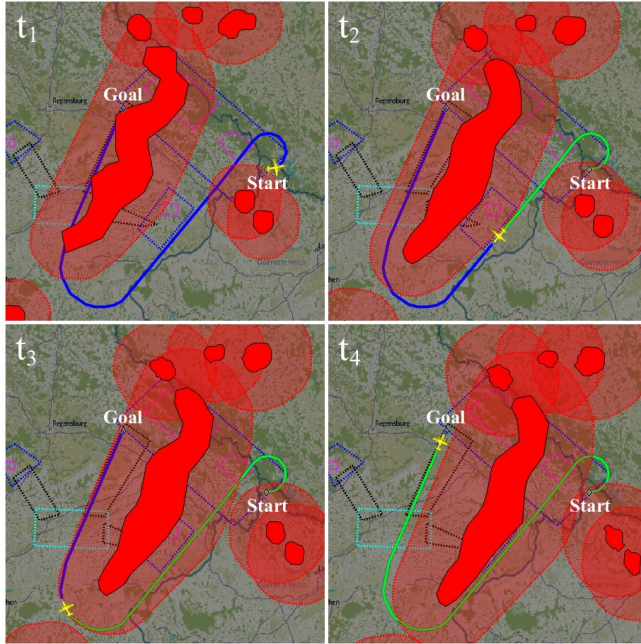
**Figure 12:** Rad-TRAM data on 2015/07/07. It is problematic if measured cells are outside of the safety margins, as they are then not taken into account by the avoidance algorithm. Positive side effect of safety margins is the inclusion of some unpredicted precipitation cells. Dashed colored lines show clusters of safety margins (one color per cluster). Small gaps between some safety margins are closed.

### Processing of No-Go areas

In this section, methods are presented to merge and expand NGAs in order to increase safety and to deal with uncertainty in the forecast. The first processing step is a spatiotemporal inter- and extrapolation. This is required for the generation of a flight path as the nowcast has a temporal resolution of 5 minutes. Due to the reasons mentioned in the previous section, most HAPS will not be equipped with a radar system. For simulative purposes, a radar system has been implemented in the module, which uses the ground based radar database from OPERA (Operational Programme for the Exchange of Weather Radar Information). Only thunderstorms inside the radar's field of view are imported (see Figure 11). Many hazards can occur in the vicinity of thunderstorms, such as gusts, wind shear, lightning, hail or icing, which is why NGAs are expanded by a minimum



safety distance. For HAPS it must be considerably larger than for an airliner. As forecasts can never predict the actual weather, it is useful to work with probabilistic safety margins. Research on the data of Rad-TRAM showed that it is possible to estimate the increasing temporal spatial deviation of a nowcast. An expansion method is applied on nowcasted cells, so that they contain their future form with a certain probability [31]. Figure 12 shows thunderstorm data (Rad-TRAM) from 2015/07/07, where polygons are differentiated as follows: ● depict the nowcast from 19:44 to 20:29 CET ( $t_0 + 45$  min), ● represent radar measured precipitation cells with identical identification number as in the nowcast, and ● show merged or unpredicted cells. It can be seen, that nowcasted and measured weather partly differs considerably. ● are the expanded NGAs (minimum plus probabilistic safety margins) at 19:44 for 20:29 CET. Over all, the inclusion of measured cells (actual weather at 20:29) is very good. The expanded NGAs leave some narrow gaps for the spatially discrete avoidance algorithm to plan through. By clustering safety margins they can be closed very effectively, resulting in an intuitive interpretation of the future weather situation.

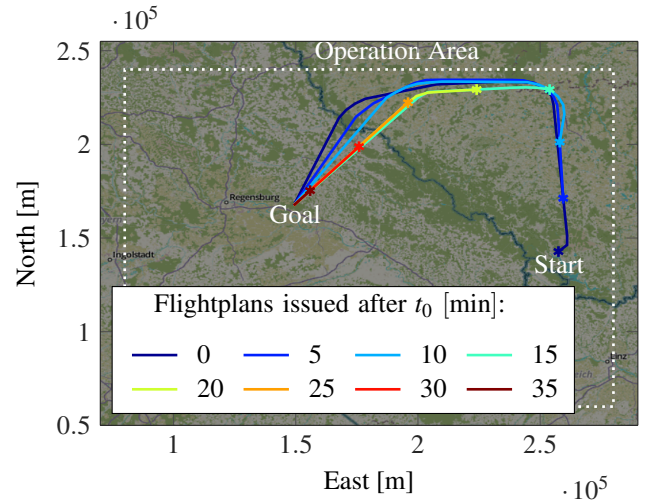


**Figure 13:** Time series of a path based on thunderstorm nowcast on 2015/06/27. Thunderstorms move from north-west to southeast. Red dotted lines show the borders of growing safety margins, due to their probabilistic component. The blue line illustrates the planned path showing no sign of pursuit curves.

#### Capabilities of weather avoidance

Flight paths can be generated inside an operation area (see Figure 14), while avoiding both static and dynamic obstacles. For this purpose, the avoidance module uses two estimator functions with different control laws, thus also allowing a prediction of the aircraft progress considering wind. This significantly improves the accuracy of the calculated paths. A lateral avoidance maneuver is always preferable over a vertical one, even for airliners. In addition, both below and above a thunderstorm, conditions are usually too rough for HAPS. With an extremely limited climbing rate, a vertical avoidance would also result in circling up for hours in order

to gather altitude to safely fly above a cumulonimbus for instance. Currently the avoidance module generates lateral paths on selectable flight levels. The algorithm finds the optimal path for each processed nowcast (see previous section). This procedure repeats with every issued update of weather data until the destination is reached or flight is aborted. Most of the time, weather develops differently than predicted so that the path has to be permanently adapted. Without radar system and solely relying on forecasts, the HAPS is in danger of suddenly being included in a newly detected NGA. A robust behavior solves this seldom problem by leading the aircraft outside and continuing the path. Figure 13 shows an example for this in the top left and also a complete flight path. The growth of safety margins with time, due to the probabilistic component is visualized with red dotted lines. The blue line illustrates the planned path showing no sign of pursuit curves.



**Figure 14:** Flight plans are updated in five minute intervals, when a new nowcast is available (indicated markers). After 37 min, the aircraft arrives safely at the goal position. Like in Figure 13, hazardous weather is avoided (NGAs of the nowcasts are omitted for better clarity).

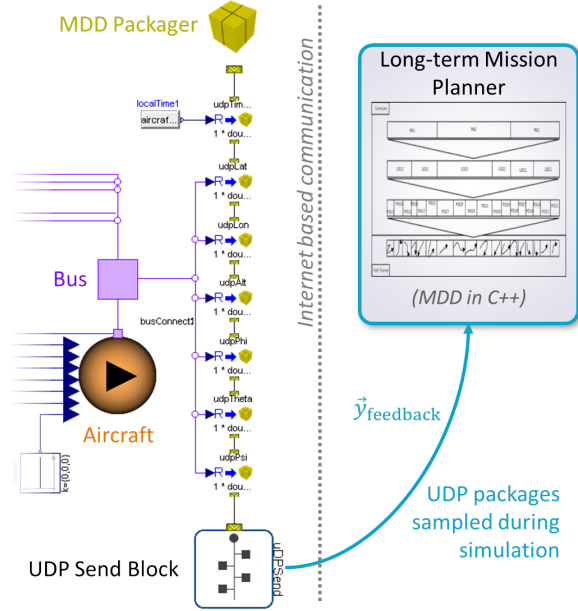
## 6. INTERFACING AND PERFORMANCE

The three main components depicted in Figure 2 were developed in different programming languages<sup>6</sup>, and on different systems<sup>7</sup>. To enable the integrated simulation, a distributed computation network had to be set up, where the interfacing consisted of file exchange and Internet-based data communication between the DLR and university, employing the UDP standard. In a real-world application, a HAPS would not receive continuous reference trajectory input, but instead calculate it onboard at hand of a trajectory data set (such as waypoints, times and tasks), which consists of a solution for several hours and is updated only periodically. This was replicated for the integrated simulation, as the planners provide such datasets to obtain a reference trajectory which can then be interpolated appropriately for the controller. Variable feedback from the aircraft to the planners however needs to be at a tighter sampling interval, which is why a UDP

<sup>6</sup>MP: C++, MATLAB, WA: MATLAB, Simulink, Simulation: C-Autocode

<sup>7</sup>MP: Intel Core i7 6700K 4GHz, Linux 64, WA: Intel Core i5 3360M 2.8GHz, Win7 64, Simulation: Intel Core2 E6850 3GHz, Win7 64

connection has been set up to model the downlink<sup>8</sup>. For this task, the Modelica DeviceDrivers library [32] was employed in *Modelica* and C++ (see Figure 15). As of now, all modules are developed and run on standard personal computers, which are different from systems that are used for onboard avionics and possibly ground control stations. Nevertheless, computation times and software in the loop tests indicate, that a combined operation is possible: The generation of a six hour optimal trajectory takes around 8 min for the long-term planner, and a one hour exemplary weather avoidance is made up of the time needed for weather data processing ( $\approx 4.2$  s), and actual pathfinding ( $\approx 28.2$  s), which depend on the amount of weather data (inside the operation area), and on the complexity of possible solutions. The DAE of the simulation is integrated by the variable step DASSL solver, taking 230 s for the combined mission scenario of Section 7, while generating around 7300 integrator events. These results confirm the feasibility of the proposed setup, for a TRL2-3 based laboratory environment. Since margins in the individual performance of the modules exist, further investigation could be motivated for example in planner parallelization, real-time capability (coding in C) for the reactive planner, followed by integration on a single computation platform or cosimulation through FMI (which has been successfully tested for instance in virtual CFD-CSM flight testing [33] and real flight testing of the *OpenInnovation/Sagitta* UAV prototype [34]).



**Figure 15:** Communication setup from simulation model (in *Dymola*) to long-term mission planner using the Modelica DeviceDrivers (MDD) library.

## 7. SIMULATION RESULTS

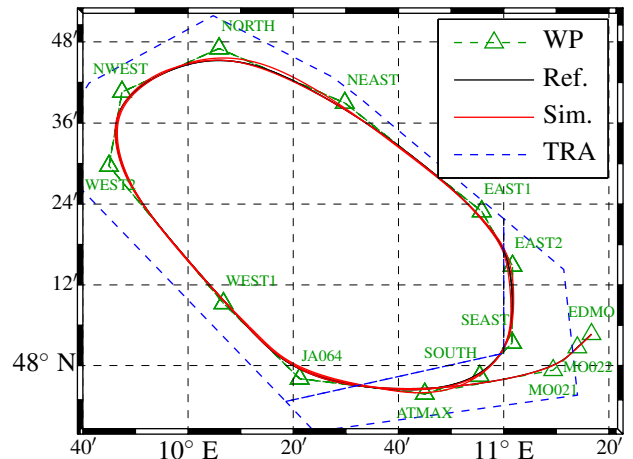
In this section, results of two simulated flight scenarios with the common date of 2015/06/27 are displayed, in which the weather data yields strong winds with high probability of thunderstorms. In reality, weather conditions would prohibit

<sup>8</sup>It is assumed that the planners run on a groundstation initially, since weather data processing is computationally expensive and requires advanced communication systems with high data rates for a safe uplink connection.

a flying clearance for the HAPS on such a day. In order to demonstrate the capabilities of the planner modules, a sufficient amount of NGAs and avoidable areas with high wind speeds are however desired. The first scenario, a climb from takeoff to the stratospheric altitude of 18 km involves passing the weather-active altitude layers, with all problematic effects already mentioned (insufficient trajectory tracking performance, risk of structural damage, etc.). To model and simulate the climb on this date anyway, the wind speed was therefore scaled to 40% of its original strength, which is deemed realistic when taking into account previous successful climbs under more favorable conditions (for instance from Airbus *Zephyr* at Yuma desert). Wind due to the turbulent kinetic energy is not scaled however, and added to the mean wind velocity. In the second scenario, the HAPS performs a typical mission scenario solely operating in the calmer stratosphere, which is why the wind strength is reset to 100% for this simulation.

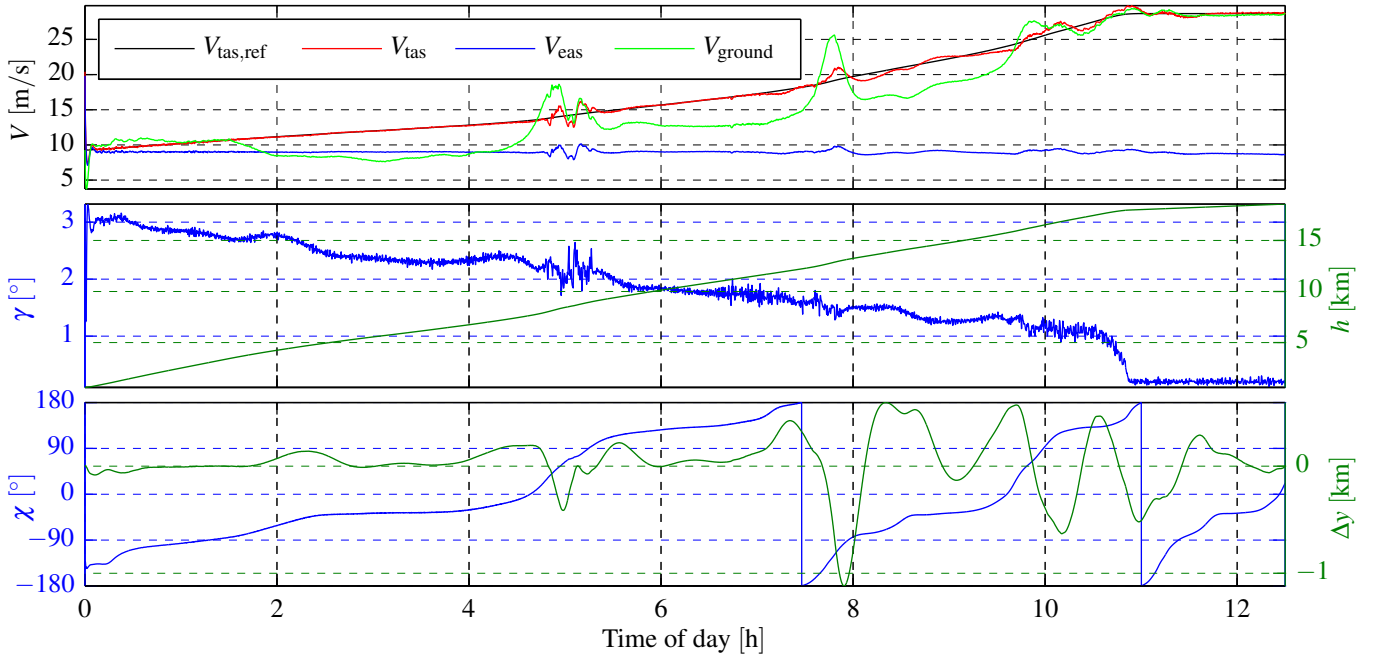
### Climb to mission altitude

Due to the previously elaborated reasons, climbing to mission altitude with the HAPS is a nontrivial exercise. During the flight through layers of the atmosphere, the aircraft experiences different strengths and directions of wind, which can often exceed the aircraft nominal cruising speed. As a simple, non-optimized climb without a mission planner involved, the aircraft should attain mission altitude in a circular pattern, ideally in an area with little or no commercial or military traffic. The Temporary Reserved Area (TRA) ED-R 307 located in the western part of Bavaria was selected as geographic constraint for the mission, since TRAs can theoretically be closed down to commercial traffic in order to conduct flight tests. The aircraft is launched from the research airfield Oberpfaffenhofen (EDMO) at 00:00 hours local time to avoid convective wind, which can be strong during daytime in the summer and for lower altitudes. After departure, the HAPS spirals up along the borders of the TRA (see Figure 16) to profit from maximum turn radii, since most of the generated lift force can be used for climbing. For operation at the point of minimal drag, a constant equivalent airspeed of 9 m/s and constant climb rate of 0.5 m/s are prescribed. The trajectory consists of a sequence of compulsory flyby waypoints forming a loop, which is traversed two times (only

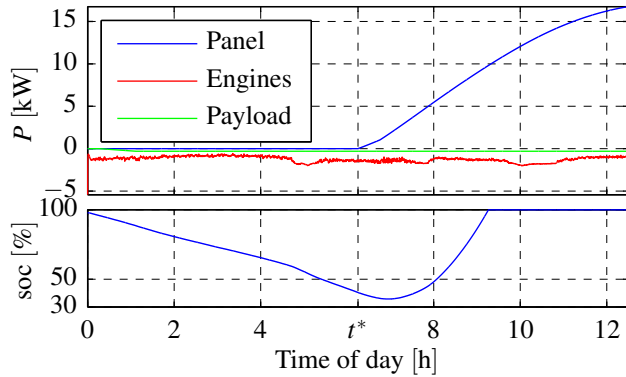


**Figure 16:** Climb trajectory mission, with the HAPS starting at EDMO and circling two times through the waypoints inside of TRA ED-R 307A/B/D (see Table 3) upon reaching the target altitude of 18 km.





**Figure 17:** Kinematical variables for the climb mission. As it gains altitude, the aircraft experiences tail- and headwinds which become apparent when comparing  $V_{\text{ground}}$  and  $V_{\text{tas}}$ . By taking into account course angle  $\chi$ , a changing wind direction during the climb can be deduced. A fixed rate of climb yields decreasing  $\gamma$ , which is mostly below a realistic value of  $3^\circ$ . The trajectory deviation  $\Delta y$  is also low, except for phases when the aircraft is driven off by wind during turning.



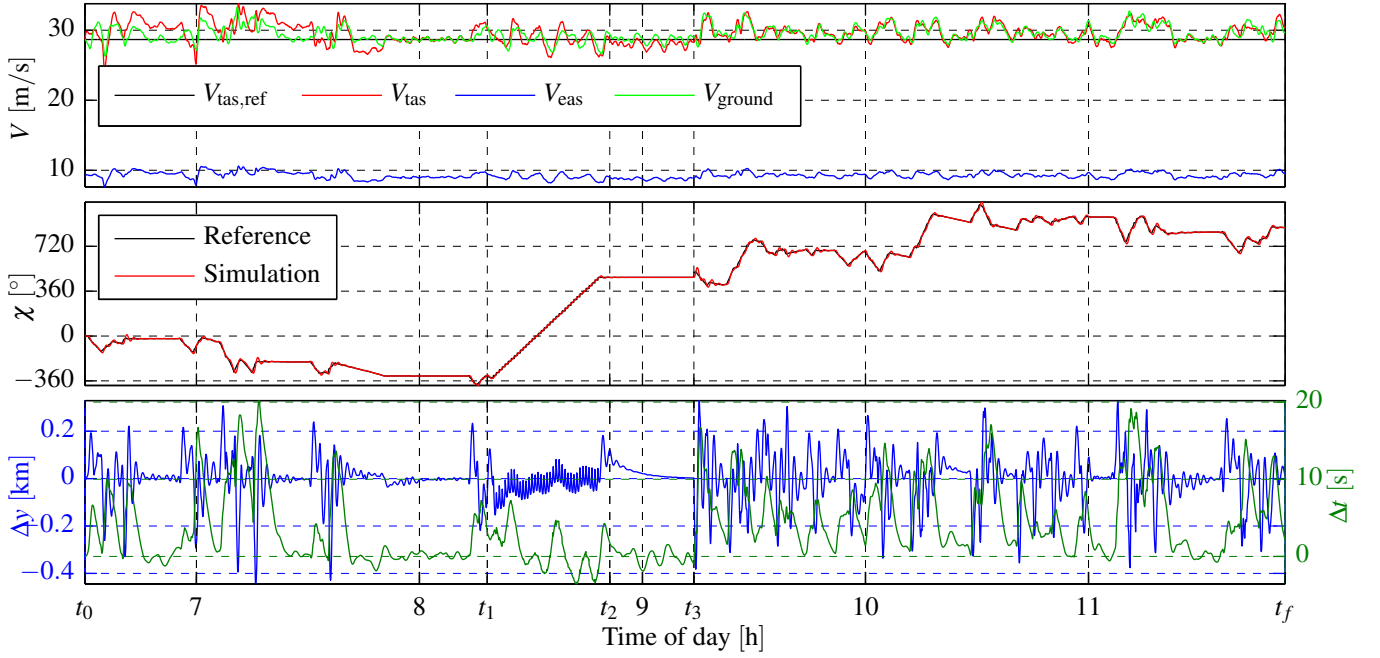
**Figure 18:** Component power outputs / demands and state of charge of the aircraft's batteries during the climb mission. The solar panel delivers energy starting from sunrise at 06:15 h ( $t^*$ ).

the coordinates of the departure and the first loop are shown in Table 3). With an initial charge of the batteries of 98 %, the aircraft drains them for approximately 6 1/4 hours until sunrise. Avionics, communication, actuator, and payload (such as a camera) power demands are modeled as a constant load of 300 W. Just before sunrise, the state of charge of the batteries drops to below 40%, as is shown in Figure 18. From this point in time, the energy retrieved from the solar panels is however high enough to quickly recharge the batteries. In Figure 17, the effects of the controller on kinematic variables is shown. It is set to control altitude  $h$ , true airspeed  $V_{\text{tas}}$ , trajectory track angle  $\chi$  and zero side slip angle  $\beta$ . In the velocity subplot, the tracking for the constantly increasing  $V_{\text{tas}}$  can be observed. The ground speed varies depending on the encountered wind, where large variations occur during

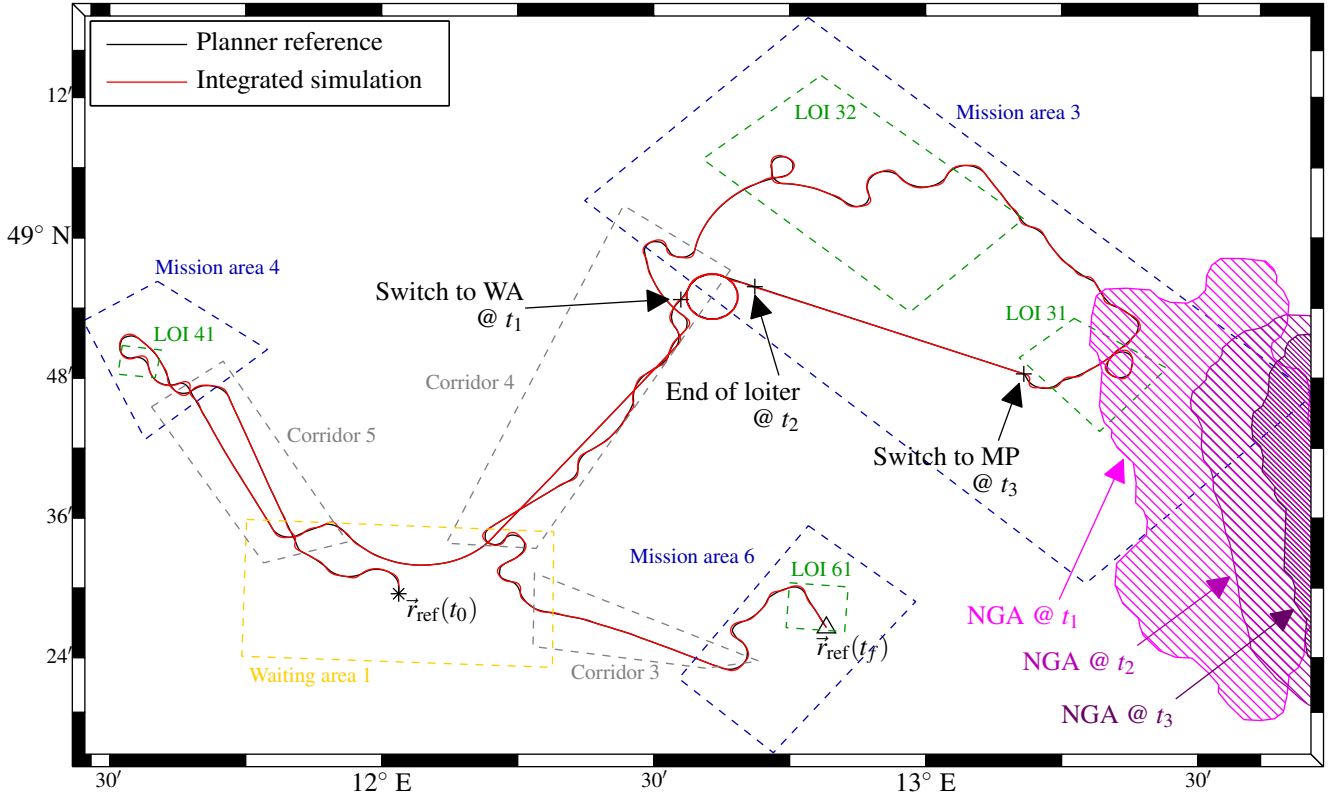
different parts of the circular flight pattern resulting in tail- or headwind. The second subplot shows the pitch angle  $\gamma$  with a realistic value of three degrees. It is constantly decreasing during the climb, as the climb rate is fixed to  $0.5 \frac{\text{m}}{\text{s}}$  and true airspeed grows. The influence of higher frequency turbulence on the pitch angle can also be seen, the target altitude is nevertheless attained. The third plot depicts the lateral deviation from the trajectory reference, where larger deviations occur during turning which can be attributed to the induced changes in wind direction and also insufficient excess power to close lateral gaps from the reference trajectory.

#### Integrated mission scenario

In the second scenario, the simulation starts on the same day at 06:30 local time. The aircraft already flies at mission altitude of 18 km, and is initially located in a waiting area designated by the mission planner. The reference trajectory is elaborated by the planner to yield maximum reward by moving to three of the six mission areas depicted in Figure 10, visiting four LOIs, and in conjunction with the WA planner, avoiding forecasted No-Go areas and strong winds while underway (see Figure 20). After the start at position  $\vec{r}_{\text{ref}}(t_0)$ , the HAPS flies to mission area 4 and visits the single location of interest. When entering MA 3, an update of RadTRAM NGAs forecast a cell of heavy precipitation over the next LOI 31. The reactive planner is activated and calculates a 4-D avoidance trajectory, taking into account the forecasted movement of the NGA in eastward direction. This new reference prescribes a loitering pattern, buying some time for the aircraft to wait until the NGA has cleared the area over the LOI. When the aircraft reaches the avoidance planner's goal point, the NGA has almost left the MA at its eastern edge and the mission can be continued. In the meantime, the long term planner has calculated a new plan starting from the WA's previously circulated goal point. The HAPS then continues to follow the new reference trajectory, until it reaches the



**Figure 19:** Kinematical variables for the integrated mission scenario. The aircraft starts at  $t_0 = 06 : 30$  h and reaches the final goal location at  $t_f = 11 : 53$  h. Between  $t_1 = 08 : 18$  h and  $t_3 = 09 : 14$  h, the weather avoidance provides the reference trajectory to avoid an NGA (see Figure 20). Reference velocity and track angle are followed with reasonable accuracy, with the lateral deviation and the time delay never exceeding 430 m / 20 s respectively.



**Figure 20:** Simulated HAPS mission scenario where the long-term planner has selected three mission areas and four LOIs to visit. Upon entering MA 3, an NGA is detected obstructing LOI 31. The weather avoidance issues a loitering pattern until the NGA has cleared the LOI. The mission is then continued with a new reference trajectory obtained from the long-term planner.

goal point at final position  $\vec{r}_{\text{ref}}(t_f)$ . In Figure 19, the velocity tracking is shown. As the planners incorporate wind in calculation of estimated arrival times at prescribed mission areas and LOIs, the 4-D reference trajectory is calculated with respect to the constraint of holding the equivalent airspeed close to the optimum value of 9 m/s. This in turn generates the desired ground track angle  $\chi$  as input to the flight control system (second subplot of Figure 19). During the loiter phase, stepwise changes are visible, which are due to the solution process in the reactive planner, treating the avoidance as a linear problem. As well, the low air density and the slow dynamic behaviour of the aircraft induce some lag and deviation in the track angle during turning. These lags also lead to deviations in the lateral distance to the reference trajectory and also in time, as shown in the third subplot of Figure 19. Maximum values of  $\Delta y_{\text{max}} \approx 430$  m and  $\Delta t_{\text{max}} \approx 20$  s are deemed acceptable for this aircraft type and application.

**Table 3:** Waypoint data for the first loop of the climb mission shown in Figure 16, where the second loop starts at waypoint ATMAX.

Name	Coord.	Track	Altitude	TAS
EDMO	N 48 04.88' E 011 16.98'	222°	610 m	9.26 $\frac{\text{m}}{\text{s}}$
MO022	N 48 02.69' E 011 14.00'	222°	875.69 m	9.39 $\frac{\text{m}}{\text{s}}$
MO021	N 47 59.31' E 011 09.40'	258°	1322.28 m	9.59 $\frac{\text{m}}{\text{s}}$
ATMAX	N 47 55.78' E 010 45.00'	278°	2878.62 m	10.38 $\frac{\text{m}}{\text{s}}$
JA064	N 47 58.04' E 010 21.21'	319°	4267.78 m	11.16 $\frac{\text{m}}{\text{s}}$
WEST1	N 48 09.27' E 010 06.73'	325°	5460.84 m	11.90 $\frac{\text{m}}{\text{s}}$
WEST2	N 48 29.65' E 009 44.97'	9°	7309.74 m	13.20 $\frac{\text{m}}{\text{s}}$
NWEST	N 48 40.60' E 009 47.45'	62°	8069.78 m	13.80 $\frac{\text{m}}{\text{s}}$
NORTH	N 48 47.02' E 010 05.90'	117°	8971.26 m	14.56 $\frac{\text{m}}{\text{s}}$
NEAST	N 48 38.97' E 010 29.80'	133°	10063.53 m	15.56 $\frac{\text{m}}{\text{s}}$
EAST1	N 48 22.92' E 010 55.83'	154°	11408.17 m	17.05 $\frac{\text{m}}{\text{s}}$
EAST2	N 48 14.82' E 011 01.68'	180°	11887.57 m	17.70 $\frac{\text{m}}{\text{s}}$
SEAST	N 48 03.37' E 011 01.60'	220°	12472.96 m	18.54 $\frac{\text{m}}{\text{s}}$
SOUTH	N 47 58.43' E 010 55.44'	222°	12790.75 m	19.01 $\frac{\text{m}}{\text{s}}$

## 8. SUMMARY AND OUTLOOK

In this work, the key components for a multiphysical simulation of a HAPS platform and typical missions are presented, consisting of a long-term mission planner, a reactive weather avoidance, a flight control system, and a detailed simulation model including environmental effects. Regarding the planning modules, mission plans and avoidance trajectories can be generated, which on the one hand respect the dynamical limits of the aircraft by avoiding adverse weather conditions, and on the other hand ensure an optimized operation with respect to the mission goals. The flight control system receives these trajectories as reference input, whose architecture is based on a combination of the total energy and total heading control approaches with standard cascaded inner loop controllers. As innermost safety layer, the controller furthermore features basic limitations for several variables,

and a pseudo-control hedging controller to consider power limitations of the electric motors by modifying reference trajectory signals. The simulation makes heavy use of the advanced abilities of the *Modelica* modeling language and introduces detailed models of the environmental conditions including 4-D weather, the aircraft systems and dynamics. Two scenarios were investigated, with the first one being a non-optimized climb to a stratospheric target altitude. By using realistic wind conditions, the mission could be completed. The second scenario implements a reference mission calculated by the long-term mission planner, involving the visit of several mission areas and location of interest in an optimal way, according to several constraints. During the flight, the reactive weather avoidance module successfully redirects the aircraft to avoid a previously unknown No-Go area emerging over one of the fly-over locations of interest. After the avoidance, the trajectory reference is switched back to a newly calculated solution of the long-term planner. The aircraft successfully completes this mission, with only minor deviations in time and lateral position tracking.

Directions for further research with respect to modeling include the consideration of structural flexibility in the equations of motion, enabling studies on stress and fatigue of the airframe, which would be highly relevant for this application. In combination, controller design and synthesis could also be subject for further studies, for example to safeguard flight and loads envelopes and ensure controller robustness to external disturbances (such as wind). Datalinks and interfacing could be further adapted to meet operational standards, and planners could benefit from performance gains by parallelization and integration on a common platform. All in all, the feasibility of a HAPS platform with increased autonomy is supported by the results of this study and motivates further research towards applications for future HAPS or HALE systems.

## ACKNOWLEDGMENTS

The authors wish to thank the Bayerische Staatsministerium für Wirtschaft, Medien, Energie und Technologie for funding the project *StraVARIA*. Thanks also go to Andreas Langmeier and Erwin Stenzel (Airbus Group Innovations) for provision of the radar simulation and help with the OPERA data, Andreas Klöckner (DLR-PD) for the work performed in management of the workpackage and modeling, Bernhard Thiele (Modelon AB) for help with the integration of the UDP connection in the simulation, as well as Martin Köhler (DLR-PA) for the support concerning weather data processing.

## REFERENCES

- [1] X. Zhu, Z. Guo, and Z. Hou, “Solar-powered airplanes: A historical perspective and future challenges,” *Progress in Aerospace Sciences*, vol. 71, pp. 36–53, 2014.
- [2] A. Klöckner, “Behavior trees for mission management of high-altitude pseudo-satellites,” Ph.D. dissertation, Universität der Bundeswehr München, August 2016. [Online]. Available: <http://elib.dlr.de/109469/>
- [3] *Functional Mock-up Interface for Model Exchange and Co-Simulation, Version 2.0*, Modelica Association, July 2014.
- [4] N. Kroll, M. Abu-Zurayk, D. Dimitrov, T. Franz, T. Führer, T. Gerhold, S. Görtz, R. Heinrich, C. Ilıc, J. Jepsen *et al.*, “DLR project Digital-X: towards virtual aircraft design and flight testing based on high-fidelity

- methods,” *CEAS Aeronautical Journal*, vol. 7, no. 1, pp. 3–27, 2016.
- [5] *Modelica - A Unified Object-Oriented Language for Systems Modeling, Language Specification Version 3.3*, 9th ed., <https://www.modelica.org/documents/ModelicaSpec33.pdf>, Modelica Association, 2012.
  - [6] A. Klöckner, M. Leitner, D. Schlabe, and G. Looye, “Integrated Modelling of an Unmanned High-Altitude Solar-Powered Aircraft for Control Law Design Analysis,” in *Advances in Aerospace Guidance Navigation and Control*. Springer, 2013, pp. 535–548.
  - [7] G. Looye, “The new DLR Flight Dynamics Library,” in *Proceedings of the 6th International Modelica Conference*, vol. 1, 2008, pp. 193–202.
  - [8] M. Otter, H. Elmqvist, and S. E. Mattsson, “The new Modelica Multibody Library,” in *3rd International Modelica Conference*, 2003, pp. 311–330.
  - [9] K. Kober and A. Tafferner, “Tracking and nowcasting of convective cells using remote sensing data from radar and satellite,” *Meteorologische Zeitschrift*, vol. 18, no. 1, pp. 75–84, 2009.
  - [10] T. Zinner, H. Mannstein, and A. Tafferner, “Cb-TRAM: Tracking and monitoring severe convection from onset over rapid development to mature phase using multi-channel Meteosat-8 SEVIRI data,” *Meteorology and Atmospheric Physics*, vol. 101, no. 3, pp. 191–210, 2008.
  - [11] M. Köhler, T. Gerz, and A. Tafferner, “Cb-LIKE-Cumulonimbus Likelihood: Thunderstorm forecasting with fuzzy logic,” *Meteorologische Zeitschrift*, vol. 25, pp. 1–19, 2016.
  - [12] M. Köhler, F. Funk, T. Gerz, F. Mothes, and E. Stenzel, “Comprehensive weather situation map based on XML-format as decision support for UAVs,” *Journal of Unmanned System Technology*, vol. 5, no. 1, pp. 13–23, 2017.
  - [13] *Product Description Document: Graphical Airman’s Meteorological Advisory (G-AIRMET)*. 3pp, Aviation Weather Center, 2010.
  - [14] L. E. Briese, A. Klöckner, and M. Reiner, “The DLR Environment Library for Multi-Disciplinary Aerospace Applications,” in *Proceedings of the 12th International Modelica Conference*, no. 132. Linköping University Electronic Press, 2017, pp. 929–938.
  - [15] M. Baldauf, A. Seifert, J. Förstner, D. Majewski, M. Raschendorfer, and T. Reinhardt, “Operational convective-scale numerical weather prediction with the COSMO model: description and sensitivities,” *Monthly Weather Review*, vol. 139, no. 12, pp. 3887–3905, 2011.
  - [16] *Flying qualities of piloted vehicles*, Military Standard MIL-STD-1797A, 1990.
  - [17] A. Klöckner, F. L. van der Linden, and D. Zimmer, “Noise generation for continuous system simulation,” in *Proceedings of the 10th International Modelica Conference-Lund*, no. 96. Linköping University Electronic Press, 2014, pp. 837–846.
  - [18] A. Lambregts, “Total energy based flight control system,” Patent US 4 536 843, 08 20, 1985.
  - [19] G. Looye, “TECS/THCS-based generic autopilot control laws for aircraft mission simulation,” *Advances in Aerospace Guidance, Navigation and Control*, 2013.
  - [20] R. Mueller and G. Looye, “A Constrained Inverse Modeling Approach for Trajectory Optimization,” *AIAA Guidance Navigation and Control Conference*, 2013.
  - [21] F. Schmitt and A. Schulte, “Mixed-Initiative Mission Planning Using Planning Strategy Models in Military Manned-Unmanned Teaming Missions,” in *Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1391–1396.
  - [22] S. Fratini, T. Nogueira, and N. Policella, “Integrating Modeling and Knowledge Representation for Combined Task, Resource and Path Planning in Robotics,” *KEPS 2017*, p. 18.
  - [23] J. J. Kiam and A. Schulte, “Multilateral quality mission planning for solar-powered long-endurance UAV,” in *Aerospace Conference, 2017 IEEE*. IEEE, 2017, pp. 1–10.
  - [24] K. Erol, J. A. Hendler, and D. S. Nau, “UMCP: A Sound and Complete Procedure for Hierarchical Task-network Planning,” in *AIPS*, vol. 94, 1994, pp. 249–254.
  - [25] V. Hethke, J. J. Kiam, and A. Schulte, “An Autonomous Mission Management System to Assist Decision Making of a HALE Operator,” in *Deutscher Luft- und Raumfahrtkongress 2017, München*. Deutsche Gesellschaft für Luft- und Raumfahrt - Lilienthal-Oberth e.V., 2017, pp. 1–7.
  - [26] L. De Filippis, G. Guglieri, and F. Quagliotti, “Path planning strategies for UAVS in 3D environments,” *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1, pp. 247–264, 2012.
  - [27] I. A. Sucan, M. Moll, and L. E. Kavraki, “The open motion planning library,” *IEEE Robotics & Automation Magazine*, vol. 19, no. 4, pp. 72–82, 2012.
  - [28] E. Scala, P. Haslum, S. Thiébaux, and M. Ramírez, “Interval-Based Relaxation for General Numeric Planning,” in *ECAI*, 2016, pp. 655–663.
  - [29] R. K. Jenamani and A. Kumar, “Bad weather and aircraft accidents—global vis-à-vis Indian scenario,” *Current Science*, pp. 316–325, 2013.
  - [30] F. Mothes, A. Klöckner, J. J. Kiam, M. Köhler, A. Pollok, A. Knoll, and A. Schulte, “Autonomes Missionsmanagement für unbemannte solarbetriebene Flugzeuge mit extrem langer Flugdauer,” 2016.
  - [31] M. Sauer, T. Hauf, and C. Forster, “Uncertainty Analysis of Thunderstorm Nowcasts for Utilization in Aircraft Routing,” 2014.
  - [32] B. Thiele, T. Beutlich, V. Waurich, M. Sjölund, and T. Bellmann, “Towards a Standard-Conform, Platform-Generic and Feature-Rich Modelica Device Drivers Library,” in *Proceedings of the 12th International Modelica Conference*, no. 132. Linköping University Electronic Press, 2017, pp. 713–723.
  - [33] R. Müller and M. Ritter, “Virtual flight testing of a controller for gust load alleviation using FMI for cosimulation,” in *Proceedings of the 12th International Modelica Conference*, vol. 132. Linköping University Electronic Press, 2017, pp. 921–928.
  - [34] R. Kuchar, “Open Innovation/Sagitta – Implementation and Validation of a Real-Time Flight Dynamics model for Simulation, Integration Testing and Pilot Training,” in *ASIM/GI Workshop*, ser. ARGESIM Report, vol. AR 53. ARGESIM Verlag Wien, 2017, pp. 61–68.

## BIOGRAPHY



**Reiko Müller** received his "Diplom-Ingenieur" degree in Aerospace engineering in 2008 from the University of Stuttgart. He is currently with the Aircraft System Dynamics department (SR-FLS) of the DLR's Institute of System Dynamics and Control. His research interests include aircraft trajectory optimization, flight guidance and control, as well as system modeling.



**Jane Jean Kiam** received in 2013 her diplôme d'ingénieur from Télécom Bretagne in France and an M.Sc. degree from the Technical University Munich (TUM) in Germany. She joined the Institute of Flight Systems at the University of the Bundeswehr, Munich, as a research assistant in 2015. Her work focuses mainly on mission planning for High Altitude Pseudo Satellite (HAPS). She is committed to developing and implementing artificial intelligence for real-world systems, especially for Unmanned-Aerial Vehicles (UAVs).



**Federico Mothes** received his "Diplom-Ingenieur (FH)" degree in Aerospace engineering in 2014 from the University Of Applied Sciences Munich. He is a PhD student in the Institute of Flight System Dynamics at Technical University of Munich. Currently he is working on the project StraVARIA in the Department of Mechanical, Automotive and Aeronautical Engineering at University Of Applied Sciences Munich as a research assistant. His research interests include system modeling, flight guidance, mission planning and weather avoidance, especially for Unmanned Aerial Vehicles (UAVs).